

Release Testing - 1.11.0

- [Testing Blocker Tickets](#)
- [Testing Plan](#)
 - [Prepare installation \(build from source\)](#)
 - [Sanity Builds](#)
 - [New Installation Tests](#)
 - [UI Tests](#)
 - [API Tests \(reference: VIVO APIs\)](#)
 - [Unload/Load Tests](#)
 - [Triplestores Tests](#)
 - [Upgrade Tests](#)
 - [Vagrant Tests](#)
 - [Performance Tests](#)

Testing Blocker Tickets

None

Testing Plan

Prepare installation (build from source)

```
git clone https://github.com/vivo-project/Vitro.git Vitro
git clone https://github.com/vivo-project/VIVO.git VIVO

cd Vitro
git checkout rel-1.11.0-RC-1

cd ../VIVO
git checkout rel-1.11.0-RC-1

cp ./installer/example-settings.xml local-settings.xml
# Update local-settings.xml with appropriate `vivo-dir` and `tomcat-dir`
```

Sanity Builds

Command	Platform	Tested by	Success?	Notes
mvn clean install -s local-settings.xml	Linux	Andrew Woods	✓	
mvn clean install -s local-settings.xml	Mac			
	Mac			
mvn clean install -s local-settings.xml	Windows	Huda Khan	✓	Used version of settings that I already had (with appropriate directories specified).
		Brian Lowe	✓	

New Installation Tests

1. Prepare installation, see above
2. Empty MySQL database

```
mysql -u root -p
mysql> drop database vitrodb;
mysql> CREATE DATABASE vitrodb CHARACTER SET utf8;
mysql> GRANT ALL ON vitrodb.* TO 'vitrodbUsername'@'localhost' IDENTIFIED BY 'vitrodbPassword';

*Note: For MySQL 8, the commands for setting up the user and granting permissions are separated:
mysql> CREATE USER'vitrodbUsername'@'localhost' IDENTIFIED BY 'vitrodbPassword';
mysql> GRANT ALL ON vitrodb.* TO 'vitrodbUsername'@'localhost';
```

3. Remove application data

```
rm -rf ${VIVO_HOME}/tdbModels/*
rm -rf ${SOLR_HOME}/server/solr/vivocore/data/*
```

3.5. Ensure runtime.properties is set up in \${VIVO_HOME}/config by copying over example.runtime.properties and specifying the URL of the Solr instance you wish to use with this VIVO deployment. Also you've copied over example.applicationSetup.n3 to applicationSetup.n3 in the same folder.

4. Start Tomcat

```
sudo service tomcat8 restart
```

UI Tests

note these are run manually, it might be possible to reconcile these test with this automated test suite : <https://github.com/vivo-community/vivo-acceptance-tests>

	Test	Platform	Tested by	Success?	Notes
1	Smoke tests do not produce error (warning about email server is OK)	Mac Linux Windows	Steven McCauley Ralph O'Flinn Brian Lowe	  	Linux Mint 19 Cinnamon
2	Home page displays correctly	Mac Linux Win/Lnx Windows	Steven McCauley Ralph O'Flinn Brian Lowe	  	
3	Log in as admin	Mac Linux Win/Lnx Windows	Steven McCauley Ralph O'Flinn Brian Lowe	  	
4	Site Admin page displays correctly	Mac Linux Win/Lnx	Steven McCauley Ralph O'Flinn	 	

5	Each function on site admin display correctly	Mac Linux Win/Lnx Windows	Steven McCauley Ralph O'Flinn Brian Lowe	✓ ✓ ✓	All 24 links (including Data Input Pull down) display normally. Ontology list All looks good
6	Site Admin - Add Individual of this class: Faculty Member (People) Project (Activities) Course (Courses) Academic Article (Research) Presentation (Events) Academic Department (Organizations) Populated Place (Locations)	Mac Linux Win/Lnx Windows	Steven McCauley Ralph O'Flinn Brian Lowe	✓ ✓ ✓	Creation of a FacultyMember without a middle name results in extra whitespace attached to the end of rdfs:label value. The default list view for publications doesn't show the issue number for articles, which seems weird, but it appears to have been this way since at least 1.9 (as with the rds:label bug above).
7	Check required fields in creating classes	Mac Linux Win/Lnx Windows	Steven McCauley Ralph O'Flinn Brian Lowe	✓ ✓ ✓	
8	Check menu entries (Home, People, Organizations, Events, Capability Map)	Mac Linux Win/Lnx Windows	Steven McCauley Ralph O'Flinn Brian Lowe	✓ ✓ ✓	
9	Check Person display (logged out, logged in)	Mac Linux Windows	Ralph O'Flinn Brian Lowe	✓ ✓	
10	Add values to all Person attributes	Mac	Ralph O'Flinn	✓	All is a heavy lift. 64 attributes accessible from the person page, some of which lead to entities (publications, positions, etc)) each with their own attributes. Recommended: Add photo, overview, URL, research areas, position. Each exercises different functionality.
11	Add research areas using vocabulary services	Mac			UMLS service requires registration key. See Release Notes. Unclear where this key would go. Other services respond as suggested. Create Jira server for this macro. It may be due to Application Link configuration.
12	Check tabs on Person display (overview, publications, contact, etc.)	Mac Linux Win/Lnx	Ralph O'Flinn	✓	
13	Add new publication to person (check required fields)	Mac Linux Win/Lnx	Ralph O'Flinn	✓	
14	Check publication sparkline	Mac	Ralph O'Flinn	✓	

15	Check manage publications (hide / show publications)	Mac Linux	Ralph O'Flinn	✓	
16	Check Publication view	Mac Linux Win/Lnx	Ralph O'Flinn	✓	
17	Check AltMetric badge (e.g. doi: 10.1038/nrmicro2331)	Mac Win/Lnx			Poor UI experience. I copied in the link, which copied in the HTML tagging, which led to showing the HTML tagging in the display. The edit form is open ended HTML, should be plain text. Should indicate what form of DOI is expected , or should accept with or without http:// without without "doi".
18	Add author(s) to publication	Mac Linux Win/Lnx	Ralph O'Flinn	✓	
19	Check Person has co-author network	Mac Linux Win/Lnx	Ralph O'Flinn	✓	
20	Check co-author network display	Mac Linux Win/Lnx	Ralph O'Flinn	✓	
21	Add grants to Person	Mac Linux Win/Lnx	Ralph O'Flinn	✓	
22	Add co-investigators to grants	Mac Linux Win/Lnx	Ralph O'Flinn	✓	
23	Check Person has co-investigator network	Mac Linux Win/Lnx	Ralph O'Flinn	✓	
24	Check co-investigator network display	Mac Linux Win/Lnx	Ralph O'Flinn	✓	
25	Load Sample Data	Mac Linux	Steven McCauley	✓	
26	Check Capability Map	Mac Linux	Ralph O'Flinn	✓	Need more sample data with more concepts for more extensive tests.
27	Check Map of Science	Mac Win/Lnx	Ralph O'Flinn	✓	
28	Check Temporal Graph	Mac Linux			
29	Search for "Derrida"	Mac Linux	Ralph O'Flinn	✓	
30	Check index page	Mac Linux Win/Lnx	Ralph O'Flinn	✓	

31	Create self editor login associated with profile	Mac Linux	Andrew Woods	✓	Updated documentation to describe how this is done. See Creating and Managing User Accounts#AssociatingUserAccountswithProfilePages
32	Log in as self editor. Verify can edit self, cannot edit others.	Mac Linux	Andrew Woods	✓	
33	Add and confirm an ORCID	Mac Linux	Andrew Woods	✓	ORCID API configuration requires clientId and clientPassword and public VIVO. See Activating the ORCID integration
34	Create editor and assign proxy editing	Mac Windows			Created user account with editor role and assigned proxy editing. Also switched to self-editor role.
35	Log in as proxy editor. Verify can edit proxied profile, cannot edit others.	Mac Windows			When logged in as self-editor designated as a proxy editor, I can only edit the proxied profile and not a different profile. When logged in as editor (i.e. the editor role which is different from self-editor), I can edit the proxied profile but also other profiles. Works as it should.
36	Check TPF endpoint (enable if necessary)	Mac Linux Windows	Andrew Woods	✓	Works as designed. 45,279 triples with VIVO data, sample data, and some manually entered test data. Design needs work. /tpf/core displayed triples and enabled triple pattern matching by subject,predicate, or object. (43,173 triples in total)
37	Run a SPARQL query	Mac Linux Windows	Andrew Woods Brian Lowe	✓ ✓	<code>curl -i -d 'email=vivo_root@mydomain.edu' -d 'password=xxx' -d 'query=SELECT ?p ?o WHERE {<http://vivoweb.org/ontology/core#FacultyMember> ?p ?o} LIMIT 5' -H 'Accept: application/sparql-results+json' 'http://localhost:8080/vivo/api/sparqlQuery'</code>

API Tests (reference: [VIVO APIs](#))

	Test	Platform	Tested by	Success?	Notes
1	Linked Open Data	Linux	Ralph O'Flinn	✓	# From Site Admin: Create a new `Faculty Member` (Add individual) ** Record the new URI: http://localhost:8080/vivo/individual/n1323 # Verify `individual` endpoint: <pre>curl http://localhost:8080/vivo/individual/n1323/n1323.ttl curl http://localhost:8080/vivo/individual/n1323/n1323.n3 curl http://localhost:8080/vivo/individual/n1323 -L -H "Accept: text/turtle"</pre> for windows using cURL for Windows 7.46
	ListRDF	Linux	Ralph O'Flinn	✓	<pre>curl http://localhost:8080/vivo/listrdf -d 'vclass=http://vivoweb.org/ontology/core#Continent' curl http://localhost:8080/vivo/listrdf -d vclass=http://vivoweb.org/ontology/core#Continent</pre>
3	Triple Pattern Fragments http://localhost:8080/vivo/tpf/core http://localhost:8080/vivo/tpf <code>curl http://localhost:8080/vivo/tpf/core</code>	Linux	Andrew Woods	✓	# Verify browsing: http://localhost:8080/vivo/tpf/core http://localhost:8080/vivo/tpf Minor observation: When TPF is disabled, it's great that the exception thrown explains how to enable it, but returning a 500 code with a stacktrace doesn't necessarily seem like the most graceful way of handling the situation. I might expect a 404 (TPF server simply doesn't exist on the site) or maybe a 501? (Request is valid, but functionality is not implemented.) The 500/stacktrace suggests to me that something is wrong with my installation. # Verify direct access: code curl http://localhost:8080/vivo/tpf/core?subject=http://localhost:8080/vivo/individual/n1323
4	SPARQL Query API	Linux Windows	Andrew Woods Brian Lowe	✓ ✓	

5	SPARQL Update API	Linux Windows	Andrew Woods Brian Lowe	✓ ✓	API works. In the course of loading approx. 90K triples via the API, this search index-related warning appeared in the log a handful of times: 2019-09-26 16:26:39,453 WARN [ConcurrentUpdateSolrClient] No more runners, but queue still has 1 adding more runners to process remaining requests on queue
6	Search Indexing API	Linux	Andrew Woods	⚠	This works in the UI, but unable to get "curl" to work. Using: curl -i -XPOST -d 'email=vivo_root@mydomain.edu' -d 'password=xxx' -d 'datafile=@uriList.txt' 'http://localhost:8080/vivo/searchService/updateUriInSearch' See: Search indexing service
7	Data Distribution API	macOS Linux	Benjamin Gross	⚠	Did not work following instructions for 1.10. Changing the DDAPI version from 1.1 to 1.1.1 DID work. Following instructions for building a snapshot version , which involved adding a repository tag, also worked. Pull request submitted to DDAPI repo to update documentation: https://github.com/vivo-community/vivo-data-distribution-api/pull/9
8	Direct2Experts API	Linux Mac			
9	Json-ld and rdf/xml individual pages				

Unload/Load Tests

	Test	Platform	Tested by	Success?	Notes
1	Load data using the Harvester	Linux			
2	Unload using jena2tools -e (using jena3tools -e Andrew Woods)	Windows Linux			
3	Load using jena3tools -i	Windows Linux			
4	Load sample data to a named graph	Windows Linux	Ralph O'Flinn	✓	
5	Remove sample data named graph	Windows Linux	Ralph O'Flinn	✓	

Triplestores Tests

	Test	Platform	Tested by	Success?	Notes
1	SDB				
2	TDB	Windows	Brian Lowe	✓	
3	BlazeGraph				

Upgrade Tests

1. Install previous release
2. Prepare installation per instructions above for previous release
3. Start Tomcat - check that smoke test passes
4. Stop Tomcat
5. Export data with jena2tools

```
java -jar jena2tools.jar -e -d /usr/local/vivo/home
```

6. Empty MySQL

```
mysql -u root -p
mysql> drop database vitrodb;
mysql> CREATE DATABASE vitrodb CHARACTER SET utf8;
mysql> GRANT ALL ON vitrodb.* TO 'vitrodbUsername'@'localhost' IDENTIFIED BY 'vitrodbPassword';
```

7. Remove application data

```
rm -rf ${VIVO_DIR}/tdbModels/*
rm -rf ${VIVO_DIR}/solr/data/*
```

8. Import data with jena3tools

```
java -jar jena3tools.jar -i -d /usr/local/vivo/home
```

9. Start Tomcat

10. Verify application per `UI Tests` above

Test	Platform	Tested by	Success?	Notes
All UI Tests	Windows Linux			
All API Tests	Windows Linux			
All Unload/Load tests	Windows Linux			
All performance tests				

Vagrant Tests

1. Clone VIVO Vagrant

```
git clone https://github.com/vivo-community/vivo-vagrant.git
```

2. Checkout release candidate

```
cd vivo-vagrant
git checkout rel-1.11.0-RC-1
```

3. Load local or [sample data](#)

4. Verify application per `UI Tests` above

Test	Platform	Tested by	Success?	Notes
All UI Tests	Linux Mac Windows			
All API Tests				
All Unload/Load Tests				
All performance tests				

Performance Tests

Performance tests should compare run times for a 1.10 vs a 1.11 on same hardware, same data. Timing collection may be facilitated by enabling [The Developer Panel](#).

	Test	Platform	Tested by	Success?	Notes
1	Compare firsttime starttime time	Linux Mac Windows			
2	Compare performance data load time				
3	Compare profile display time				
4	Compare SPARQL query time				
5	Compare Search Index rebuild time				
6	Compare Inferencer rebuild time				
7	Compare jenatools export time				
8	Compare jenatools import time				
9	Compare Data Distribution API response time				